

Projektisuunnitelma; Leevi Pulkkinen, 910776, Energia- ja ympäristötekniikka, 2020, 25.2.2021

Tarkoituksena toteuttaa simppeli tasohyppelypeli, jossa pelaaja ohjaa hahmoa, joka liikkuu kentällä. Kentän rakenne muodostuu kentän aloituspisteestä, ”vihollisista”, kerättävistä kolikoista ja maalista, josta siirrytään seuraavaan kenttään. Alustavasti olisi tarkoitus suorittaa projekti vaikeustasolla VAATIVA.

Ohjelma suorittaa tarvittavan kommunikaation pelaajan kanssa nuolinäppäinten ja hiiren välityksellä. Kun peli on käynnissä, pelaaja pystyy liikuttamaan pelihahmoa nuolinäppäimillä vasemmalle ja oikealle, sekä hyppäämään. Pelin kentän näkyvät kokonaisuudessaan luodussa ikkunassa. Tarkoitus olisi myös lisätä jonkin tapainen main menu, josta voi valita kentän tai sulkea pelin. Main menussa ei näy kenttää vaan kaksi nappia, ALOITA ja POISTU. Näitä nappeja voi käyttää klikkaamalla hiirellä.

Esimerkki: Pelaaja avaa pelin ja päätyy aloitusnäyttöön. Pelaaja klikkaa hiirellään ALOITA napista, jolloin aloitusnäyttö muuttuu pelin ensimmäiseksi kentäksi, jossa on hahmo, jota pelaaja voi ohjata. Eli ALOITA nappia painaessa ohjelma luo ennalta luodun tason ja piirtää sen näytölle. Pelaaja liikuttaa hahmoa kentässä ja osuu viholliseen. Pelaajan hahmo palaa takaisin kentän aloituspisteeseen. Koko pelin ajan ohjelma jatkuvasti tarkistaa osuuko pelaajan ohjaama hahmo vastustajiin, ja jos osuu, pelaaja palautetaan tason alkuun. Pelaaja pääsee tason loppuun ja uusi taso alkaa. Pelaaja sulkee pelin. Jos pelaaja pääsee tason maaliin, piirtää ohjelman seuraavan tason ruudulle välittömästi. Jos tasoja ei enää ole, pelaajan ilmoitetaan voittaneen pelin. Kun pelaaja sulkee ikkunan, peli sulkeutuu.

Luokat:

Peli:

Ohjelman avautuessa piirtää näytölle aloitusruudun ja ALOITA nappia painaessa luo kaikki osat mitä pelin pelaamiseen tarvitaan. Peli luokka on ohjelman pääluokka ja kaikki muut luokat polveutuvat tästä luokasta. Peli luokassa tapahtuu myös eri olioiden välisten törmäysten tarkastus.

Hahmo:

Hahmo on pelinappula, jota pelaaja liikuttaa antamalla komentoja nuolinäppäimillä.

Taso:

Taso luokka huolehtii eri tasojen luomisesta. Taso luokka piirtää saamansa datan mukaan ruudulle tason, jossa pelaaja pelaa.

LiikkuvaVihollinen:

Eri tasoilta löytyy liikkuvia vihollisia. Jos pelaaja törmää hahmolleen viholliseen, alkaa taso alusta ja pelaaja ”häviää”.

StaattinenVihollinen:

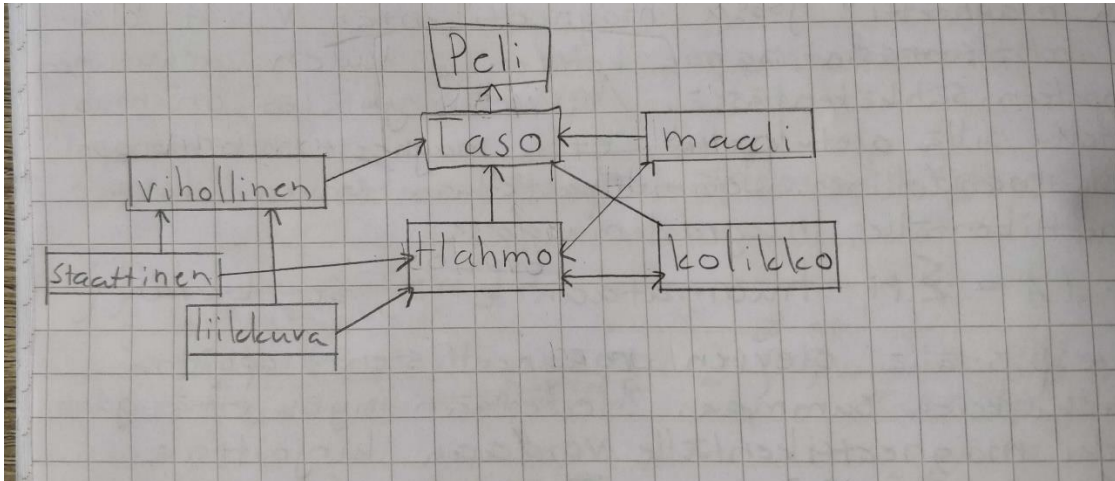
Vihollinen, joka ei liiku mutta toimii muuten samalla tavalla kuin edellä mainittu LiikkuvaVihollinen. Voisi olla esimerkiksi piikki tai vaikka laavaa.

Kolikko:

Tasolta löytyvä esine, joita voi kerätä törmäämällä. Kun kolikkoon törmää, se katoaa ja pelaajan pisteet nousevat yhdellä. Jos pelaaja häviää, pisteet nollautuvat.

Maali:

Kun pelaajan hahmo törmää maaliin, pääsee pelaaja seuraavalle tasolle tai voittaa pelin.



Kaavio luokkien vuorovaikutuksista.

Tällainen luokkarakenne tuntuu luonnolliselta ja riittävän yksinkertaiselta toteuttaa.

Ohjelman tärkein tietorakenne tulee olemaan kaksiulotteinen lista, joka kuvaa pelin tasoa. Ajatuksena olisi jakaa luotu ikkuna ruutuihin ja jokaista ruutua kuvaa yksi kaksiulotteisen listan kohta. Tämä tapa on hyvä, koska merkkiaamalla kaksiulotteisen listan alkioita eri arvoilla voidaan määrittää mitä kyseiseen ikkunan ruutuun tulisi piirtää. Esimerkiksi listan alkio voisi olla  $[0][2] = 0$  eli sen on tyhjä ja  $[3][5] = 2$  on esimerkiksi vihollinen. jne.

Ohjelma voisi hyödyntää tiedostoja tasojen rakanteen säilyttämiseen ja mahdolliseen tallentamiseen, jos tasoeditori toteutuu. Kenttien rakenne olisi tiedostossa kaksiulotteisena listana

Ohjelma ei tule tarvitsemaan mitään monimutkaisia algoritmeja. Ainut mitä tarvitaan, on törmäyksen tunnistus hahmon, tason pintojen, vihollisten, kolikkojen ja maalin välillä.

Ohjelman tärkeimmät testauksen kohteet ovat hahmon liikkuminen ja sen törmäminen ympäristöön. On tarkistettava, että hahmo ei pääse poistumaan tason alueelta, se toimii oikein, kun se törmää vihollisiin tai kolikkoihin ja, että yleisesti pelin fysiikka vaikuttaa järkevältä. Näytä asioita on oikeastaan helppoa testata pyörittämällä ohjelmaa ja pelaamalla peliä, ja samalla tarkastelemalla toimivatko kaikki asiat halutulla tavalla. Myös datan, joka luo tason, on oltava oikean laista, jotta tason luomisessa ei aiheudu ongelmia.

Näillä näkymin ainut tarpeellinen kirjasto on PyQt5, koska sillä hoidetaan ohjelman graafisen puolen toteutus.

Ohjelman kasaaminen saattaisi olla paras aloittaa yksinkertaisen tason rakentamisella ja sen piirtämisellä näytölle. sen jälkeen lisätään ohjattava hahmo ja tason ja hahmon välisen törmäyksen tunnistus. Sen jälkeen yksi kerrallaan lisätään viholliset ja kolikot jne. Myöhemmin voidaan muodostaa lisää tasoja, lisätä pistesysteemi tai tasoeditori ja muut tarvittavat ominaisuudet.

Ohjelman kasaamisen aikataulua on todella vaikea arvioida, koska sen valmistumisnopeus riippuu paljon siitä, kuinka paljon aikaan ohjelman koodaamiseen pystyy laittaa. Uskon kuitenkin, että jo maaliskuun aikana on tehty jo hyvää edistystä ja uskon pelissä olevan yksi taso ja hahmo, jota voi siellä liikutella tarkoituksenmukaisella tavalla. Huhtikuussa voi sitten lisäillä uusia ominaisuuksia luodun rungon päälle.

Olen tutustunut etukäteen youtubessa löytyneisiin videoihin, joissa on esitelty itse tehtyjä tasohyppely pelejä, mutta niissä kaikissa on käytetty pygame moduulia. Sain näistä videoista kuitenkin hyvän kuvan siitä, millaisten teemojen ympärillä tämantapaisten pelien luominen pyörii.

